AIAA 2002-1054

The TENOR Architecture for Advanced Distributed Learning and Intelligent Training

C. Tibaudo, J. Kristl and J. Schroeder
Ontar Corporation
North Andover, MA

B. Sorensen
USAF AFRL/HEA
Mesa, AZ

D. MacCuish
Air Command and Staff College
Maxwell AFB, AL

40th Aerospace Sciences Meeting & Exhibit
14-17 January 2002
Reno, Nevada

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
JAN 2002		

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
The TENOR Architecture for Advanced Distributed Learning and Intelligent Training	**F33615-00-M-6033**
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER **65502F**

6. AUTHOR(S)	5d. PROJECT NUMBER
Cheryl Tibaudo; Joseph Kristl; John Schroeder; H. Barbara Sorensen; Don MacCuish	**3005**
	5e. TASK NUMBER **HA**
	5f. WORK UNIT NUMBER **0E**

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Ontar Corporation, 9 Village Way, ,North Andover,MA,01845	

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This paper describes TENOR - the Training and Education Network On Request. TENOR is a multi-media, adaptive training system, in which the content and rate of presentation is paced by the capabilities and requirements of the user. The web is an ideal media for providing an adaptive training environment where the material presented to every individual is generated "on-the-fly" and is tailored to the individual's particular needs and capabilities. The entire United States Department of Defense community is actively engaged in distance learning, and all the US military services are making a commitment to it. This also holds true for a growing number of commercial and academic institutions worldwide, which have realized the importance and economy of utilizing ADL and the web for education, training, information sharing and performance support. The TENOR software advanced distributed learning (ADL) system has several primary advantages over other systems. This paper will examine the details of the system architecture and its database fields, the lessons and modules developed, customization abilities, and unique features such as cross-platform use, links to real-time information, audio capabilities, student-instructor interaction options, and automatic bandwidth adjustments. It will also illustrate how a student will access, logon, and progress through a lesson.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**10**	

2002-1054

Abstract

This paper describes **TENOR** – the **T**raining and **E**ducation **N**etwork **O**n **R**equest. TENOR is a multi-media, adaptive training system, in which the content and rate of presentation is paced by the capabilities and requirements of the user. The web is an ideal media for providing an adaptive training environment where the material presented to every individual is generated "on-the-fly" and is tailored to the individual's particular needs and capabilities. The entire United States Department of Defense community is actively engaged in distance learning, and all the US military services are making a commitment to it. This also holds true for a growing number of commercial and academic institutions worldwide, which have realized the importance and economy of utilizing ADL and the web for education, training, information sharing and performance support. The TENOR software advanced distributed learning (ADL) system has several primary advantages over other systems. This paper will examine the details of the system architecture and its database fields, the lessons and modules developed, customization abilities, and unique features such as cross-platform use, links to real-time information, audio capabilities, student-instructor interaction options, and automatic bandwidth adjustments. It will also illustrate how a student will access, logon, and progress through a lesson.

Background

The Ontar Corporation has developed a system architecture and prototype software that implements a unique solution for the training of force protector teams for the US Air Force, using advanced distributed learning (ADL) techniques. This solution is called TENOR, which stands for the Training and Education Network on Request. TENOR uses a unique approach to implementing training over the internet to provide anytime-anywhere training on desktop, laptop or notebook computers and Personal Digital Assistants such as the Palm Pilot and Handspring units. TENOR is a multi-media, adaptive training system in which the content and rate of the presentation of material are paced by the capabilities and requirements of the user.

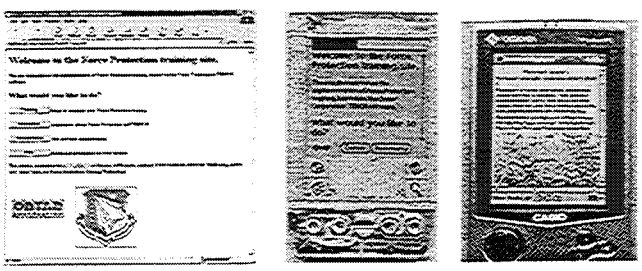
This paper is divided into two sections: the TENOR architecture and its development, and the Force Protection Module. Ontar has been working on a Phase I Small Business Innovative Research contract with the United States Air Force Research Laboratory to design and develop an Advanced Distributive Learning, or as we call it, an intelligent training system. It utilizes the enormous capabilities of the web, PDAs, and traditional desktop technology. Development will continue under a Phase II SBIR contract, with work anticipated to start in a month or two. This ADL system is called TENOR, for Training Education Network on Request. There have been a number of recent learning systems developed that leverage off Internet technology. TENOR uses a unique approach to implementing training over the Internet. Now it is being more fully developed and implemented, using the latest wireless technology and several testing partners.

It was designed to solve training problems for the specific situation of Force Protection training. Currently the Air Force has bases and other facilities protected by the Army. In the future, the Air Force will have its own so-called "mini marines" to protect the bases. The resulting architecture and tools of TENOR are suitable for all distributed learning tasks, for both military and commercial (schools and industry) users. As a matter of fact, an end goal of the SBIR program is commercialization.

The TENOR software architecture is designed to deliver training over the Internet on just about any topic that can be presented in a browser-based interface. The software highlights include a database-driven storage of knowledge content and lesson formatting, the ability to adjust lessons to match the limitations of the student display platform, and the capability to match the level of content to each individual student's requirements and capabilities. The software has been developed and tested on the specific training task for Force Protection personnel, but the architecture and approach is general enough to handle a wide range of training and educational tasks. We are currently working with several US Government and public academic institutions to develop additional country modules. We are also working with private commercial businesses to develop medical training modules and leisure travel modules.

Currently, web based ADL is not taking advantage of the many features that are unique to a web environment. For example, we know that all trainees do not advance at the same rate, or even that all the trainees preparing for an operation require the same level of training. Supervisors require "big picture" information, while the workers need the "nuts and bolts" material. The web is an ideal media for providing an adaptive training environment where the material presented to every individual is generated "on-the-fly" and is tailored to their particular needs and capabilities.

Furthermore, the means by which we receive information is rapidly changing due to the tremendous advances in technology. In the not too distant future, the capabilities of handheld computers, such as the Personal Digital Assistants (PDA's) using the Palm® or MS Windows CE® operating systems will rival those of current desktop machine. The highly mobile, wireless features of these devices make them ideal platforms for anytime-anywhere ADL. At this time, these PDAs present several challenges. For instance, they have less "real estate" or screen size than desktop machines and bandwidth is currently limited to 14.4 baud modems. The TENOR system accomplishes its tasks by NOT using static web pages, but by storing training material in a database and creating customized web pages on the fly.



As is becoming increasingly known and accepted, Advanced Distributed Learning, or ADL, is a method

of providing instruction to students who are not physically located at the training facility. It can be as simple as viewing an instructional video obtained from your local public library to the more complex situation of a multimedia, interactive web site. In the past, ADL has been implemented using methods based on a variety of technologies, including training materials presented via mail, videotape, CD-ROM and closed-circuit television. The explosion of Internet technology and software tools provides a method of delivering high quality content almost instantly to remote locations. Current Internet standards can be used to deliver text, graphics, and multimedia audio and video elements into widely available browser software. This promises to revolutionize the delivery of distant learning and has particular advantages for presenting information that is rapidly changing or that must adapt to the requirements of different trainees.

ADL for Training

There are several advantages to an Internet-based implementation of Force Protection training. First, the Internet provides easy access to training for remote users. There is no need for an instructor to be present and available during the training, and no facilities are needed beyond an Internet connection and a host browser. Second, an Internet based solution provides the ability to rapidly update or add courseware information, to match changing conditions and requirements. Third, a browser-based courseware provides the opportunity for a user to interact with the system and customize the training to his or her level of expertise or an interest in the subject. Fourth, current standards allow the same solutions to be used in internal networks for those topics or material that should not leave the organization. These are compelling advantages over traditional methods of distance learning that make an internet-based training solution worth a close look.

The web is an ideal media for providing an adaptive training environment where the material presented to every individual is generated "on-the-fly" and is tailored to their particular needs and capabilities. The entire US Department of Defense community is actively engaged in distance learning. For example, at the Air Force Command and Staff College at Maxwell Air Force Base, Alabama, over half the trainees are "off site." Similar numbers are reported by most of the United States military's training organizations. All the US military services, and many of those worldwide, are making a commitment to distance learning. Military officials believe ADL will augment recruiting and retention efforts and help develop a more educated, informed and techno-savvy military.

ADL is also becoming an important component of civilian academic institutions. One recent report finds that 7 in 10 US colleges now offer some form of online learning, including courses, lecture notes, and online study groups. That number is increasing daily. Many institutions are also offering two- and four-year degrees entirely via computer. Commercial firms are also increasing their use of ADL for training and other distributed learning tasks. The TENOR software ADL system developed has several primary advantages over other methods of ADL previously developed. They include:

platform flexibility, which allows TENOR to support multiple platforms by simultaneously formatting the training material for different browser systems, ranging from "traditional" desktop computers to Personal Digital Assistants such as the Palm and Windows PocketPC;

trainee level of expertise, which enables the material presented to the trainee to be created "on the fly." Consequently, the lesson is easily tailored in real time to accommodate the capabilities of the individual trainee;

trainee customization, means the system is customized to the individual trainee throughout his DoD career. As the individual advances, changes jobs, etc., the system tracks the trainee's requirements for future training needs;

training material, which is decoupled form the system as much as possible via an intelligent interface. The experts who implement the training material can easily compose their lessons using text based, audio or video material from a remote site without any need to know or understand the structure of the training material database;

Lesson updates, in which the training material is easily updated via the interface and database structure, and finally,

Flexibility of training material. The TENOR system can accommodate any type of training material, including engineering, maintenance, medical, weapon handling, etc. An expert in any area can interact with the system via the web and create training material for any kind of application.

TENOR achieves its objectives using three general concepts. First, is the server based software. The

server hosts the ADL software on a single server, operated and maintained by the training provider. Students connect to this server using standard Internet browser software, which simplifies student requirements. Use of a server provides security and flexibility in the development and updating of training material. The lessons are created in HTML, a widely used standard. In addition, the lessons provide support for generating Javascript language scripts, which adds considerable flexibility and capability to the presentation of lesson material.

Database driven lessons – TENOR defines an architecture where the material that makes up each lesson is stored in a series of database tables. Lesson material is stored in a knowledge database that can consist of text, graphics, or multimedia elements. Additional database tables define the layout of each lesson screen. The lesson screens are generated by the server software when requested, so instead of a static set of lesson material, the screens are dynamic and thus easily updated. Fields are provided to assign bandwidth requirements and expertise level information to each element, which the software can use to customize lesson presentations for each student's ability, requirements, and browser platform.

Decoupling of content and presentation – a unique advantage of the TENOR architecture is that lesson content and presentation are maintained in separate database tables. This allows lesson displays to be customized to match the limitations of platforms of interest (such as small PDA screens), and also lets multimedia elements be supported on more capable systems. The lesson content creator can provide unique versions for every platform, or a single default version that runs on all platforms.

The Force Protection training module is supposed to provide relevant information for Force Protection personnel about to deploy to a foreign location. The training can be accessed from any computer connected either into the Internet or into an internal DoD server. This allows the training to be accomplished rapidly and efficiently.

The Force Protection module developed in Phase I consists of 1 module with 10 topics, encompassing about 60 screens. There are 4 overview lessons that first must be completed by the student. These lessons provide general instruction that help the students know more about themselves and how they might interact with other members of a Force Protection team. The lessons are summarized briefly as follows.

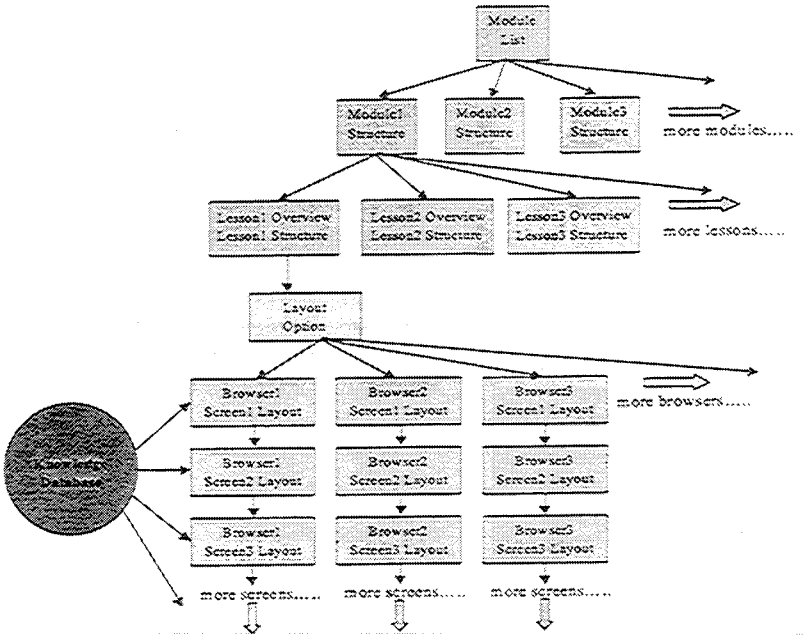
AOR Model of Learning – this lesson discusses the way that students learn from experience. It provides a useful review of how the student must pay attention to the lessons of experience, both in this training and throughout life.

Knowing Oneself – this lesson provides access to some personality tests and lets the student discover what kind of person he is, and how he is likely to interact with others. Since an important component of Force Protection involves human interaction (with other members of the team, as well as with the local authorities and population), this helps the student understand personal strengths and weaknesses when going into this job.

Air Force Core Values – this lesson provides a summary of Air Force Core values. Since Force Protection personnel are likely to come in contact with representatives of foreign countries as well as news media operations, it is valuable to review Air Force core values so that on these potentially visible missions, personnel can conduct themselves in a manner consistent with Air Force standards.

Humanitarian Interventions – this lesson provides an overview of non-governmental organizations that get involved in providing humanitarian assistance abroad. Many Force Protection missions are expected to occur in tandem with humanitarian efforts by these organizations, which may or may not be coordinated with the US government. After completing this lesson Force Protection personnel will be aware of the various organizations, as well as their sponsors and agendas.

After completion of these mandatory top level lessons the student can then select the country or area where the Force Protection training will occur and learn more about it. The overall goal of the "region" training is to provide background information, context, and useful information to a Force Protection person working overseas. These lessons are not scholarly treatises on the country or region in question. The training thus focuses on items such as history, economy, government structure, country infrastructure, roads, facilities, and language. The Phase I effort developed a prototype training module for a single country: Montenegro. A fully developed Force Protection training module would provide training lessons on many countries and regions, so that airmen deploying to any location could get accurate and useful familiarization with the place in which they will be working.



A number of unique features capitalizing on the ability of browser computers were built into the Phase I lessons. For example, a link is provided to an internet service that gives the current weather in the capital of Montenegro. This allows an interested student to check the weather as frequently as he is interested, and start to get an intuition of the current conditions there. Another unique feature added in is audio narration of a number of scenes, created with Ontar's consultant to this project, Don MacCuish. The audio is loaded as a background sound on computers that are capable of playing it during the lessons, adding to the effectiveness of the presentation. For systems with insufficient connection bandwidth (such as a modem connection), the audio is not sent so that the lesson is not bogged down in data transmission delays. These provide a sampling of the powerful options that can be supported in lessons hosted in the TENOR software.

As other countries or regions are added as additional modules to the Force Protection training, they will likely not follow this lesson structure exactly. Each country around the world has unique features that are important for Force Protection personnel to be aware of, and training must be customized for those areas.

Database and Features Overview

The basic TENOR system has two parts. First a relational database is used to store both the basic knowledge elements and its structure into lessons. Secondly, a server based program accesses the various databases and builds the training lessons on the fly as the student works through them. The sever software is coupled with a relatively small static site to handle initial connections and provide features that change infrequently and do not need to be created on the fly. The internet browser is used as the client platform for the learning system. The lesson presentation uses existing HTML standards for display text and multimedia information, for maximum compatibility across browser software and indeed for platform independence. This approach combines a wide range of display and interaction capability with existing off-the-shelf solutions to provide a powerful system at greatly reduced costs.

Using a series of database tables to store both knowledge and formatting information provides several advantages over the "hard-coding" of lesson presentation and material. First, it provides some separation between the knowledge content and the display of that knowledge. This allows the sever flexibility in adapting the presentation to the capabilities (and limitations) of the various possible browser platforms that a trainee might use. TENOR is implemented as a DLL that runs on a server with either Microsoft Internet Information Server 4 or 5 (IIS), or the Microsoft Personal Web Server (PWS). The server also requires the establishment of an appropriate ODBC database DSN so that the database tables can be accessed within the software.

Central to the TENOR system is the definition and use of the databases used to store the learning content and lesson structure. Ontar made a conscious decision early on to use standard relational database technology to implement this feature. There are many COTS database products that can serve this need, with capabilities ranging from limited to truly enormous. To limit the constraints imposed by selecting a proprietary database system (and then being locked into its use on the server), the software interface to the databases was designed to use the Open Database Connectivity (ODBC) convention. This convention provides a generalized API to the database content, at the cost of some performance on

the server end. The advantages of database vendor-independence were judged to outweigh performance issues. In concept any relational database software that has an ODBC interface can be used for the TENOR databases.

The database tables are accessed within the code using the ODBCDirect object model within the Microsoft DAO (Data Access Objects) framework. The tables are registered as ODBC data sources on the server, and the resulting DSN (Data Source Name) string provides access to them through ODBC drivers installed in Windows. A series of data structures are defined within the TENOR server code, and when they are needed, separate subroutines have been written to populate the structures with the current values from the tables. Connections established to ODBC databases are used and then removed as quickly as possible, since maintaining open database connections can have a large impact on server performance.

In TENOR, the student must select which module to use, and then which lesson within that module to work on. The lists of possible modules and lessons are generated from those currently defined in the TENOR database tables ("topic list" and "topic structure" tables). Only modules that apply to the current student (as determined from the student database entries) are displayed, and the lesson lists are modified by the progress so far of the student. Once a lesson is selected, the server software loads the lesson definition tables first ("lesson overview and "lesson structure" tables for that lesson). The browser and platform type is determined, and then the browser type table ("layout option" table) is used to select the appropriate tables for screen layout definition ("screen structure" table). For every screen in the lesson, the screen elements are loaded from any of the knowledge databases ("knowledge element" tables) and formatted using the selected "screen structure" table. Status is maintained on where the student is and has been, so that backward and forward screen motion can be accomplished.

At the end of each lesson is an evaluation test(s) to ensure that the student has mastered the material to an acceptable degree. It is possible to define a lesson that includes zero tests, or to have more than one test. Since the student interface is based on a browser, test questions use standard HTML input and display elements to pose problems and collect student input. A pool of questions can be maintained and actual tests can be generated from random selections from the pool, so that the tests are not identical when repeated. Test inputs are stored, and the designer of the lessons can input criteria to apply to the inputs so that some type of scoring can be implemented. TENOR also supports a non-interactive, or "lecture hall", presentation mode for each lesson. In this mode the server, not the student, controls the pace of presentation of the material. The forward and backward buttons are removed from the lesson interface screen. However, the state of the "Interact" and "Exit" buttons are checked during the display of any screen, to allow the student to interrupt the presentation either for a question or to terminate it early.

Five general types of database tables are defined in the TENOR system: knowledge database, the lesson structure tables, the lesson presentation tables, test databases, and the student database. TENOR is started by using a browser to launch the TENOR.ASP file located at the root of the TENOR directory created on the host server. For the Ontar server hosting TENOR, this means that the browser uses http://10.0.0.2/tenor/tenor.asp as its starting URL. The student then logs in with a user name and password. He will see 6 buttons, which provide access to generic support functions, including forums and help. Once the student has selected a training module to work on, the screen is displayed. This is done by TENOR first querying the database tables that describe the selected module to determine what lessons it contains, and the list of lessons is displayed on the screen. Lessons that have already been completed by the student are highlighted with an asterisk. Once the student clicks on the "Begin Lesson" button, he can start working. Now the student can interact with the server while taking a lesson. He can: get help; move to any point within a lesson; replay and review a segment of the current lesson; compose a question and send it to an "instructor"; exchange comments and questions with other students; terminate the lesson; search the TENOR knowledge database for additional material; take a test; configure the defaults for the training session, based on the student's current computer parameters; resume a lesson; bookmark the lesson; exit the lesson. Embedded audio tracks narrate the content on the screen, if the browsing platform has the correct hardware and necessary bandwidth. Even simple handheld PDA units can effectively present text and graphics to the user.

One component of the TENOR software is its relatively small Web-site front end. This collection of static HTML pages handles the tasks that do not change significantly over time, and thus it makes more sense to create these pages as relatively static HTML than to regenerate them constantly. This front

end handles the initial connection of the trainee into the system. It also works with the server software to determine the characteristics of the host platform, including the browser type, the platform display capability, and the connection bandwidth. The user can either specify this directly, or some simple software tests can be run once to determine these parameters. The parameters are stored with the student and re-used for future training sessions, with an ability to always override the stored defaults. The front end handles establishing a secure connection if necessary, and provides the infrastructure to host the help for the system, as well as the discussion forums provided to allow students to discuss the lessons with each other and exchange relevant information.

The front end also collects demographic and other support information on the student, through a series of input screens. This information allows the performance of the trainee to be tracked and provides the hooks for the software to customize presentation of material. Approval is being obtained through appropriate US Air Force channels for collection and safeguarding of this information, and suitable protections against any compromise are being developed.

Central to the TENOR system is the definition and use of the databases used to store the learning content and lesson structure. Ontar made a conscious decision early on to use standard relational database technology to implement this feature. There are many COTS, or commercial off-the-shelf, database products that can serve this need, with capabilities ranging from limited to truly enormous. To limit the constraints imposed by selecting a proprietary database system, and then being locked into its use on the server, the software interface to the databases was designed to use the Open Database Connectivity (ODBC) convention. This convention provides a generalized API to the database content, at the cost of some performance on the server end. The advantages of database vendor-independence were judged to outweigh performance issues. The availability of Java versions of this interface, called the Java Database Connectivity, or JDBC convention, was also a strong positive feature that encouraged adoption of a database convention with ODBC-based software access. In concept, any relational database software that has an ODBC interface can be used for the TENOR databases. In the initial Force Protection training system, the Microsoft Foxpro database provided adequate capability and was thus used to create that system.

The server software is the heart of the TENOR system, and it accomplishes several essential tasks. First, it handles the lesson interface, screen by screen, to the trainee. Once a lesson is initiated, the student browser begins a direct dialog with the server software. The server programs generate each screen of each lesson on the fly, first by determining from the lesson databases what the next appropriate screen is, and then extracting the knowledge elements and layout information and using this to build an HTML page and hand it back to the trainee browser. The HTML code is customized to the capabilities of the host browser, as the course creator did when populating the knowledge and layout databases. The server software also responds to input from the trainee, if any, and then either generates pages in response to the trainee input or continues on with the lesson. It also tracks student performance and stores results within the trainee database for later use.

Since the lesson display tool in TENOR is a browser, the presentation of information is organized into a series of screens. Screens can consist of static information such as text and graphics built from the database, or it can include multimedia elements as well. Any valid browser form input element, including button, text box, list input, etc., can also be displayed anywhere in a lesson screen, thus providing the ability for the lesson to have custom branches, separate hyperlinks, or nested levels of information. The actual layout and presentation of each screen is customized for the browser capability of the student, which is determined at the start of a student's session. One primary task for the server software is to determine, based on the student's inputs, what the next screen is that should be displayed. Lessons are assembled in a preferred order, but TENOR provides the ability to randomly access other screens within a lesson, or to exit at any time. Since the screens are being built on the fly, the accurate determination of the next appropriate screen is very important.

An expert at a remote site creates the training material using an "Authoring tool", which will look and feel to him very much like Microsoft Word or Power Point. TENOR massages the expert's material and puts it in the database for future use as needed.

Field Name	Field Use
kn_name	name assigned to this knowledge element
kn_type	type of information in this element
kn_content	actual content, or pointer to content (depends on kn_type value)
kn_keyword	keywords that apply to this knowledge element
kn_source	source this content was taken or derived from
kn_related	names of other related knowledge elements
kn_comment	comments on this item added by the knowledge expert

This brings us to the "soul" portion of TENOR. There are three general types of databases that are implemented in the TENOR system. First is the knowledge database. This contains the knowledge, or raw material, that will be used in the training lessons. It is decoupled as much as possible from the layout and presentation of the learning material. The knowledge is broken up into a series of knowledge elements, which can range from bullets of text displayed on a screen to documents or multimedia elements. The knowledge database either directly holds this information or contains pointers that direct the software to the location of the information. To assist in searches and other database manipulations of material stored external to the database, keyword fields for material are maintained.

The second type of database is the layout, or lesson, database. This database first organizes elements of knowledge information into screens, which are the basic unit of display within this browser-based learning system. The position and attributes of every knowledge element displayed are stored in the database, and different values for the layout and display are maintained for different browser platforms. This is how the lesson presentation can be generalized to support a wide range of client system capabilities. Additional tables within this database then order the screens and related learning material into a coherent lesson. Tests to evaluate the comprehension of the material, and to ensure that the trainee has mastered the material to an acceptable level, are also maintained within this database and are associated with lessons. Support is also provided to store lesson objectives and other related lesson material.

These two databases are very complex and can grow quite large even for a moderately complicated lesson. Development has started on a series of graphical user interfaces that will allow experts to populate these databases with meaningful knowledge and lesson material, without having to be database software experts. This software operates in two modes. In the first, the expert can create screens, or lessons, on a storyboard software display. Text, graphics, and multimedia elements can be inserted, moved, and organized as required. The software extracts both the knowledge elements and the layout information from the lessons created by the expert and populates the database with the appropriate values. The software then provides options such as previews on all possible platforms, from desktop to PDA systems, and over a range of bandwidth settings, so that the course designer can evaluate the results and adjust them as needed for optimum effect. In the second mode, existing lessons or knowledge elements are displayed and the expert can edit them or create new lessons or tests from them. A complete range of functions required to support a lesson, such as the development of tests and the establishment of completion criteria, can also be edited and associated with each lesson. When this interface is completed it will provide a powerful extension of the TENOR system to allow support for training of arbitrary topics.

The third type of database maintained in the TENOR system relates to the trainee characteristics and performance. Any basic demographic information about the student that is needed is maintained here, along with a record of training completion and status. This allows the software to authenticate a student's identity and limit access to the system, which is an important consideration for something exposed to the open Internet. It also provides the baseline data needed to track student progress. This allows the software to give both trainees and supervisors feedback on progress. Software additions to TENOR are under consideration to allow reports of progress to be generated and e-mailed directly to trainees and supervisors. A supervisor interface is also in development that will allow lists of required and recommended training lessons, for each student, to be stored and then used when that student connects to the system and accomplishes further training.
Now to discuss the current Force Protection module that has been developed in TENOR.

Force Protection Module

Rapid deployment of competent military teams throughout the world requires operationally trained personnel to be prepared to manage all threats and to employ all contingencies. Currently in the US Air Force, these force protector teams are comprised of individuals who differ in competency skill levels in the fields of medicine, intelligence, communication and security. In addition, the force protectors assigned to a team generally originate from different geographical locations and have varied operational missions and experience levels. To further complicate the US Air Force force protector team mission, individual team members likely are unfamiliar with each other, have no detailed knowledge of the deployment location, and are not trained in the specifics of the tasks to be performed. These troops, however, must be provided with the maximum training and protection available. Travel budgets do not allow for all force protectors to have access to the most recent and complete training and equipment, and troops are not able to gather at a

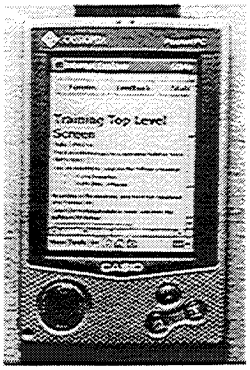
central location to learn about an assignment. Yet, training has to occur almost constantly, as modern technology means that changes in required information occur more rapidly and frequently.
These requirements underscore the need for training that begins at the initiation of an assignment and continues throughout arrival at the theater of operation. Recent trends favor accomplishing this necessary training through advanced distributed learning methods. ADL methods and technologies provide alternative capabilities to train rapidly deployed force protectors. ADL allows for formal, institutionally based training, despite the instructor and student being geographically separated.
The TENOR software architecture was initially developed to implement a course in Force Protection training for the US Air Force. In Phase I of our work, the outline of the training was developed, and the basic architecture was established. Initial versions of much of the software were developed and debugged. We are currently working in a Phase II effort. The software architecture and tools are being completed and tested on a more elaborate version of the Force Protection training.

The Force Protection training consists first of a series of independent lessons grouped into an initial module. This module is a precursor to taking any of the country or region specific training material. These lessons remind the student about *HOW* learning takes place, provide some feedback as to understanding how the student might react in certain situations, and provide general information on relevant Air Force core values and humanitarian aid organizations. The four lessons implemented in the current version of TENOR are:

#1 AOR, or Action, Observation and Reflection Model of Learning,
#2 Knowing Oneself,
#3 Air Force core values, and
#4 Humanitarian Interventions.

Some of these lessons are presentations of situations, and highlight a number of examples to get their point across. Other lessons simply present material to remind the student of fundamental concepts, such as the lesson on Air Force core values. The final lesson, Humanitarian Interventions, provides an overview of the types of organizations, such as the Red Cross, that personnel involved in Force Protection tasks may have to interact with while in the field.

Once the initial module is completed by the student, he can then access a series of modules that provide in-depth information on specific countries or regions that the student may be deployed to. In the Phase I effort, we selected to develop course material around Montenegro. In the Phase II effort these courses are being expanded to include more depth and more material relevant to the personnel that could be deployed in a Force Protection situation. The eventual plan is that this system will be expanded to include many countries or regions, and the courseware material will be developed by experts on those regions, using the various TENOR software tools.



Each country module consists of a series of lessons that present background material to provide context for personnel deployed in a Force Protection mode. This includes topics like the country location, geography, history, currency, demographics, and cultural and ethnic groups. The goal is not to provide a comprehensive scholarly presentation on any of these topics, but rather to extract what is needed for personnel who will spend some time in the country and very likely have extensive interactions with the local people and government.

Conclusion

In summary, Ontar is implementing an INNOVATIVE ADL system under the US Air Force SBIR program. The highlights of the system are one: that learning material is stored in a database, and NOT in static web pages; two, our intelligent engine adapts to the trainee's level of expertise, learning style and task requirements; three, it supports multiplatform formats including desktop and wireless PDAs; and four, it can be used for any type of training, such as medical, force protection, automobile repair, farming - or any application that requires training. Ontar Corporation has completed an extensive investigation of the best methods for hosting an ADL system using internet technology. This has resulted in the development of TENOR. TENOR is a unique, advanced database system that achieves the objectives of providing timely, cost-effective training, while using the latest innovations in desktop and wireless computer technology.